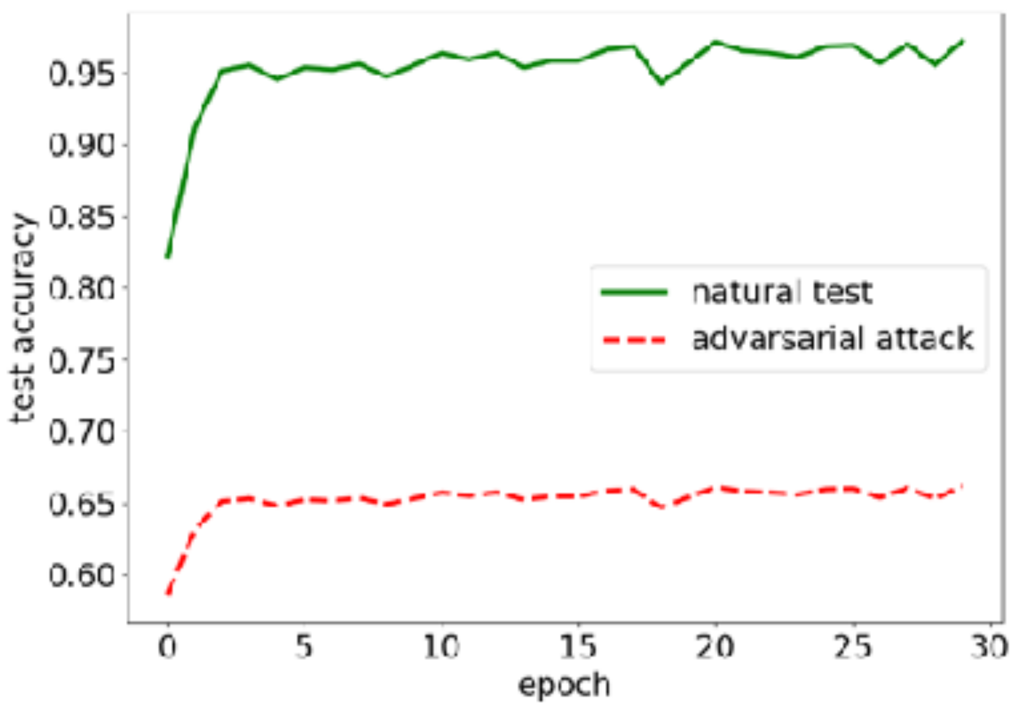
Improving the Robustness of Neural Network via Data Augmentation

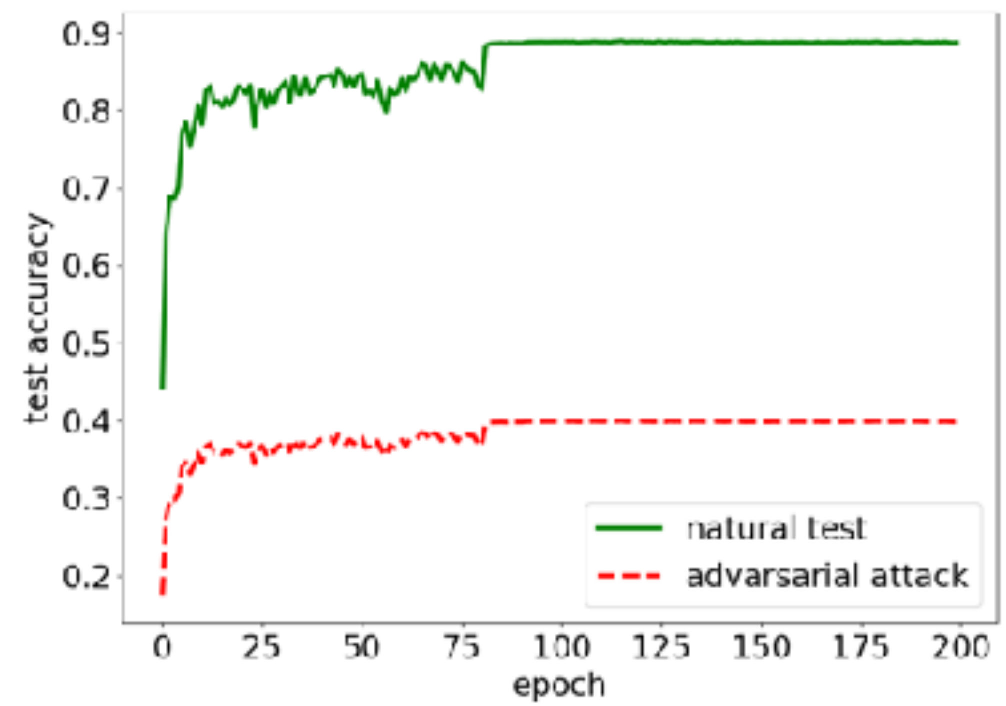
12/13/2018

Problem definition

- Robust generalization is quite different from standard generalization[1]
  - Some data sets may not large enough to train a robust model
- Neural Network can be fooled with simple spacial transformation (rotation, translate)[2]



GTSRB



Cifar10

Existing approaches

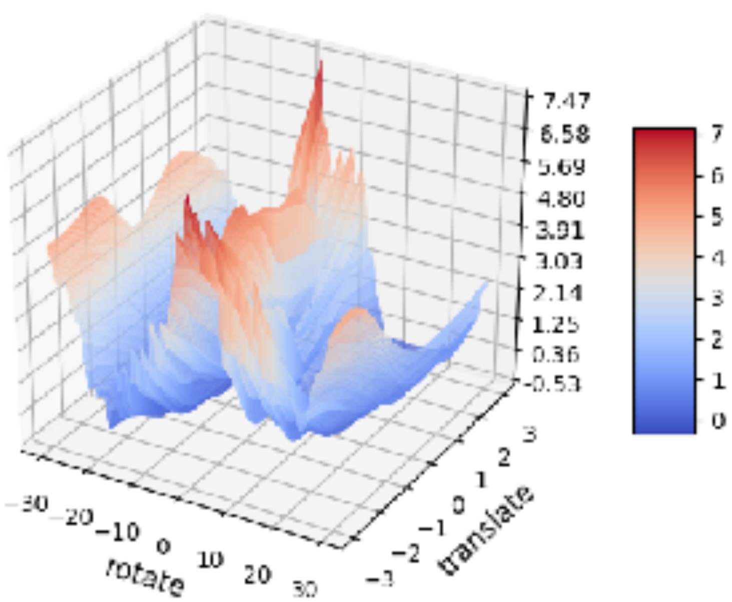
- **Adversarial learning**: existing strategy tries to perturb the training data set in each step with the goal to learn the features of potential adversarial variances
  - random perturbation
  - Worst-of-n: select the most representative images from randomly generated n perturbations
- **Limitation**: input space is too large, especially with more transformations. Random perturbation may not be able to find representative images.

For GTSRB, even though start-of-the-art model archives 98% test accuracy, we can generate adversarial example for more then 17% with rotation range $(-30^{\circ},30^{\circ})$ and translation range(-3p, 3p).

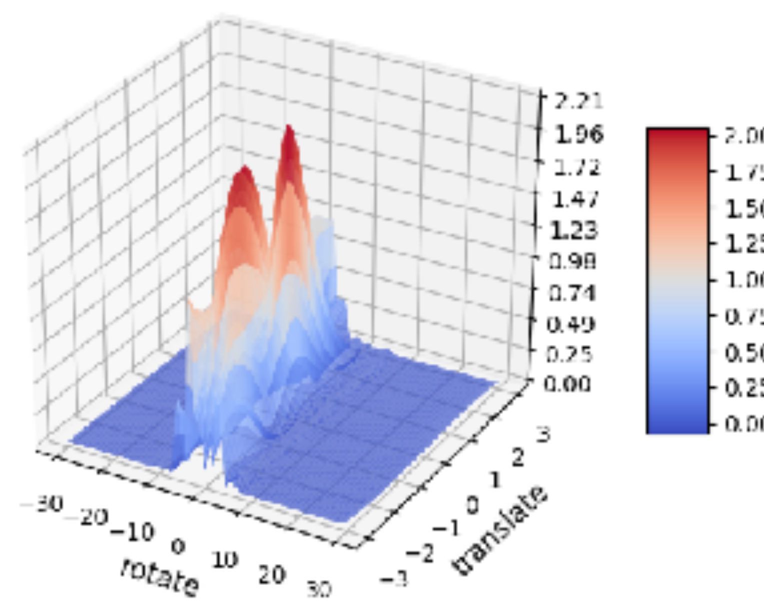
Adversarial examples

Rotate 1°

One adversarial example from GTSRB dataset. The model trained using worst-of-10 approach still misclassifies the perturbations (rotate 1°).



traditional model



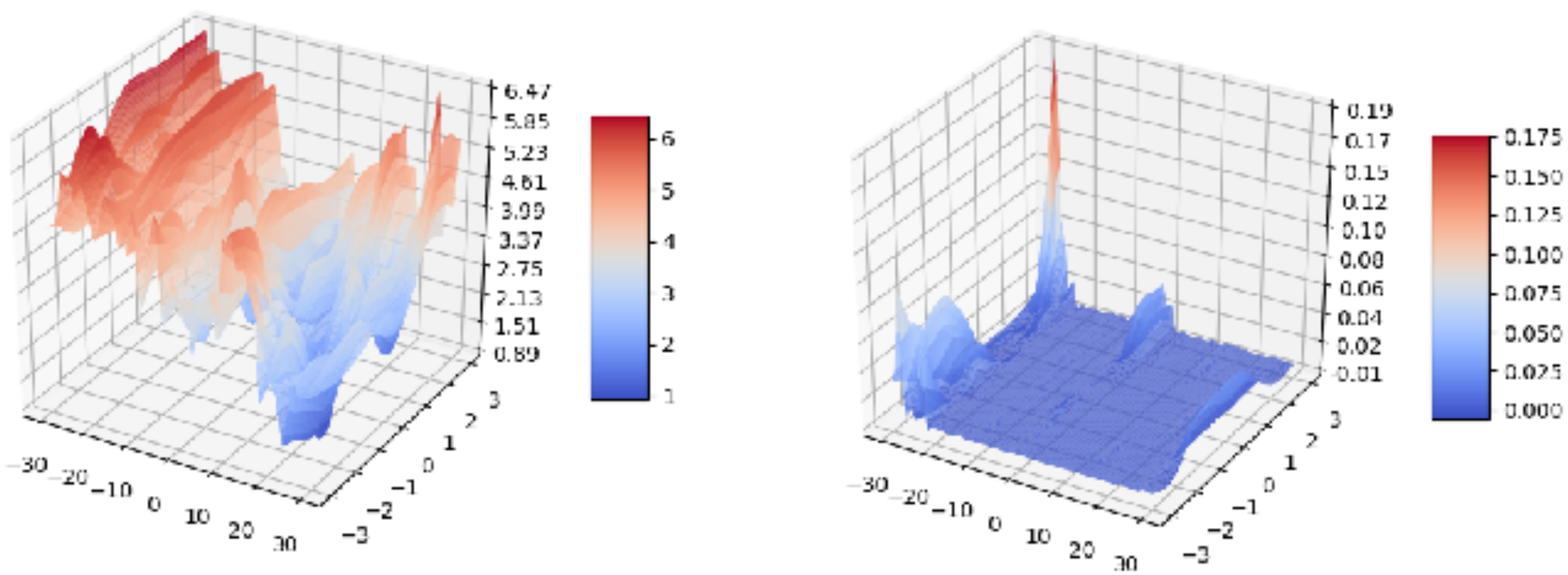
worst-of-10 model

The loss of perturbed images with rotation (-30, 30) and translate (-3, 3)

Better, but not good enough

Challenges

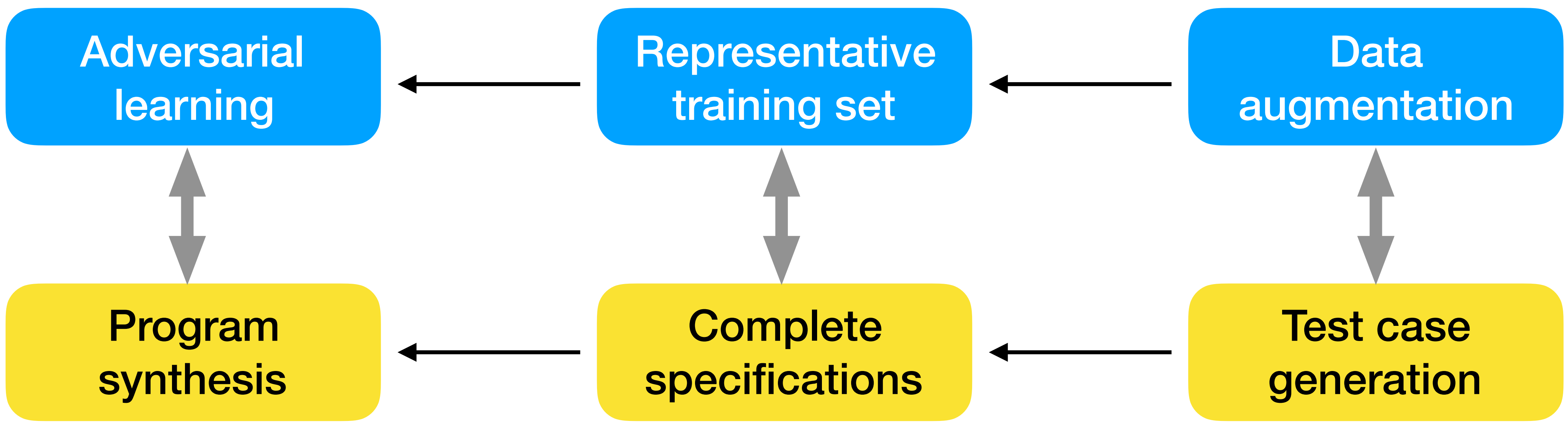
- Input space is too large, especially with more transformations, computing all adversarial variances is a time-consuming and sometimes impossible task
- Input space is unstructured (non-concave maximization), gradient-based approach cannot directly applied



The loss of two examples with rotation (-30, 30) and translate (-3, 3)

Intuition

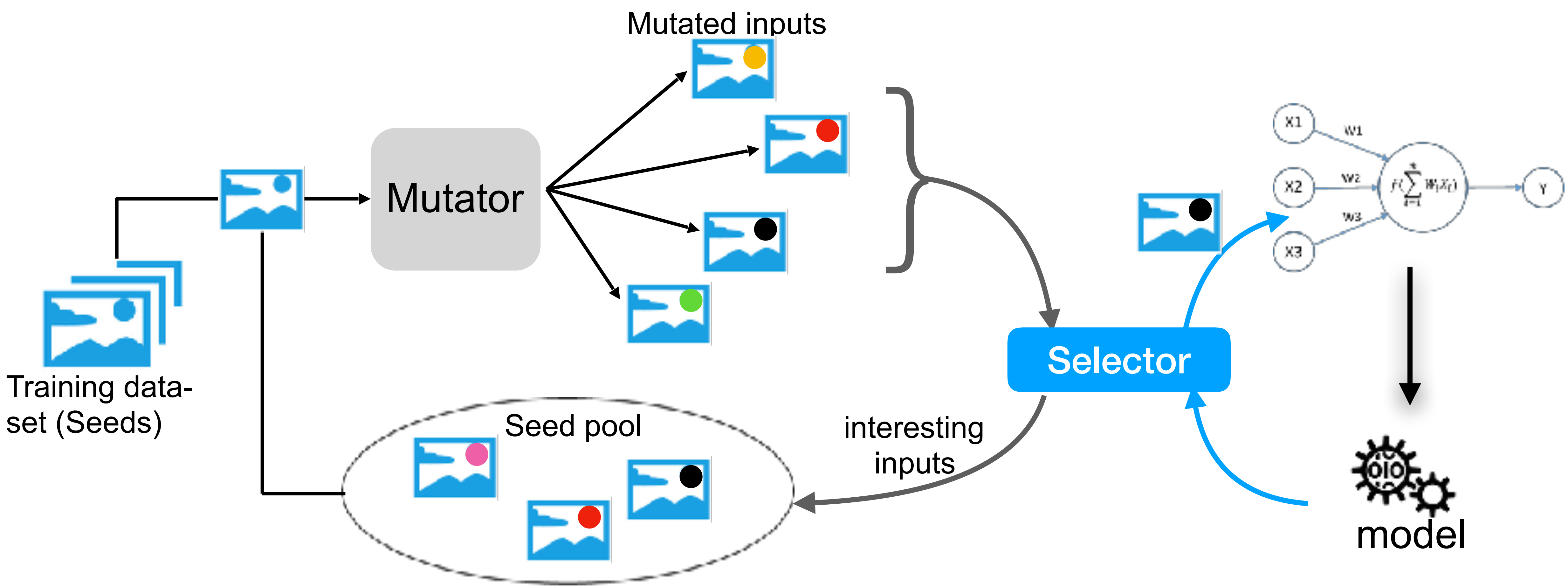
- Model training can be regarded as AI-based program synthesis process. Given a set of specifications, it will generate a program satisfying all the specifications



- Data augmentation can be used to provide more complete specifications
- We can formalize representative training data generation as a **search problem** within the attack space
- If one perturbation is misclassified, its neighbours are more likely to be misclassified

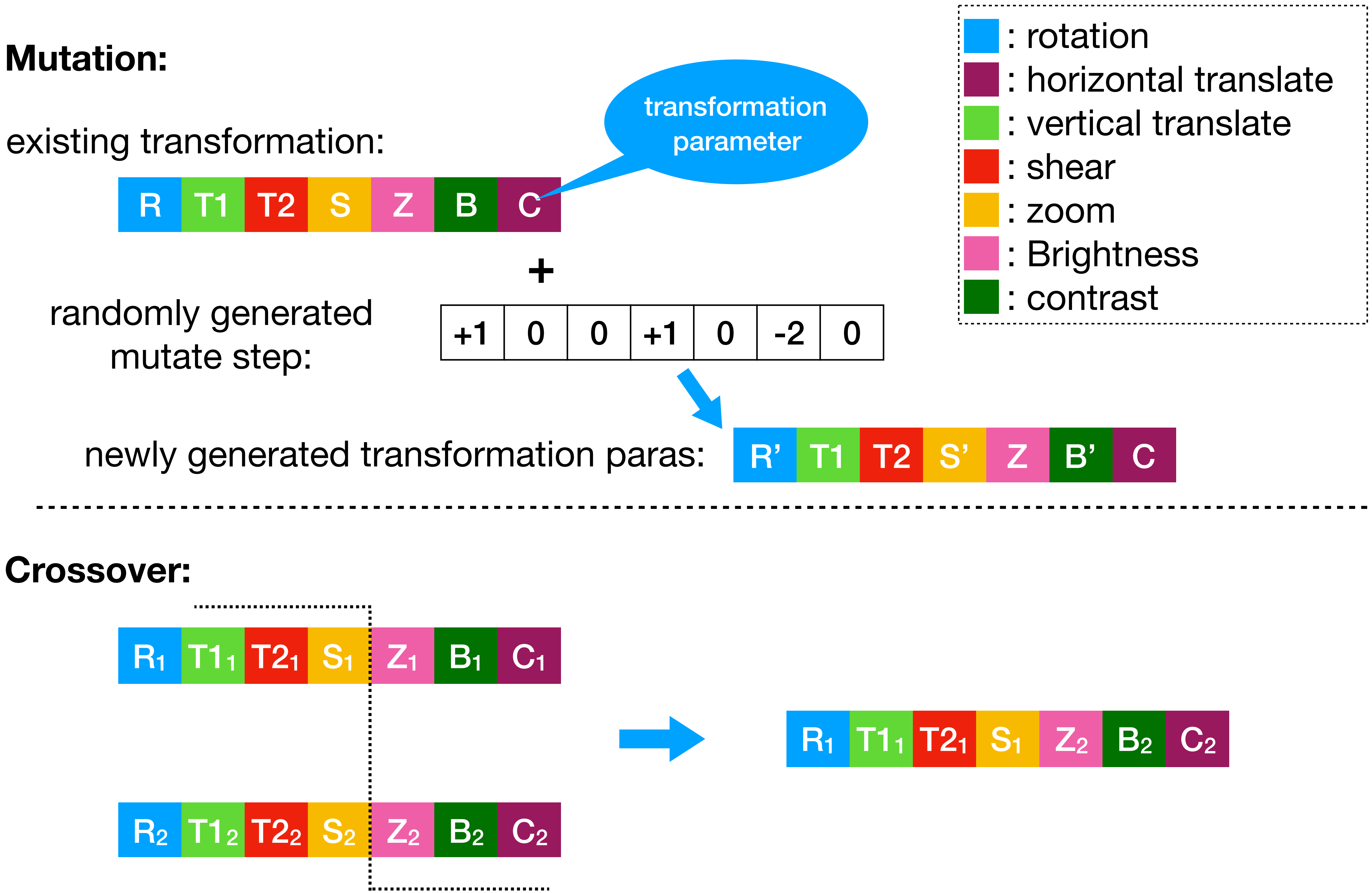
Overall workflow

- Using genetic algorithm to generate representative perturbations.
- The goal is to maximise the diversity of samples in the distribution



- Mutator generates new perturbations based on existing seed
- Selector selects interesting perturbation and continually maintains the seed pool

Mutator

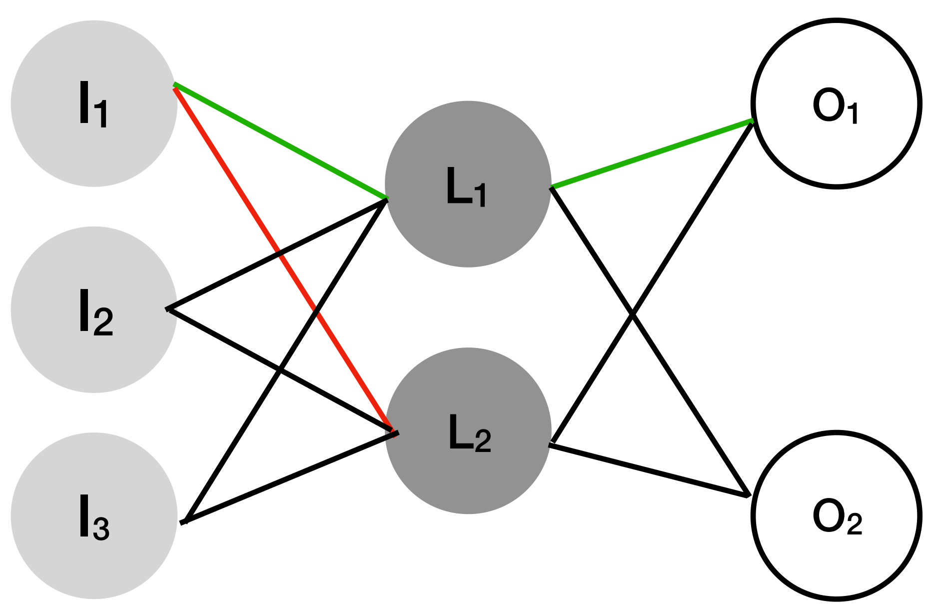

Mutation:
existing transformation:
R T1 T2 S Z B C
transformation parameter
+
randomly generated mutate step:
+1 0 0 +1 0 -2 0
newly generated transformation paras:
R' T1 T2 S' Z B' C
: rotation
: horizontal translate
: vertical translate
: shear
: zoom
: Brightness
: contrast
Crossover:
R_1 $T1_1$ $T2_1$ S_1 Z_1 B_1 C_1
R_2 $T1_2$ $T2_2$ S_2 Z_2 B_2 C_2
R_1 $T1_1$ $T2_1$ S_1 Z_2 B_2 C_2

Fitness function

1. Loss-based approach
   - Saddle point problem $\min_{\theta} \mathbb{E}_{x} \left[\max_{\|x'-x\|_{\infty} \leq \varepsilon} \text{loss}(\theta, x') \right]$

     Where x is original data, and x' is the perturbed data
   - Prefer perturbation with higher loss value (categorical cross-entropy)

2. Neural coverage-based approach (ongoing strategy)
   - Take the model structure into consideration
   - Prefer perturbation that can improve neural coverage



For instance, suppose inputs covering *I1* and *L1* are correctly classified, in the next step, we prefer inputs that cover *I1* and *L2.*

Evaluation

- Dataset:
  - GTSRB: German Traffic Sign Benchmarks with 50,000 images and 43 labels
  - Cifar10: Cifar-10 Benchmark with 60,000 images and 10 labels

- Data augmentation strategy:
  - Standard: model trained based on original training data
  - Aug.30(40): model trained based on randomly perturbed training data, 30 (40) is the perturbation parameter range
  - Worst-of-10: randomly generating 10 perturbations for each image, and train the model using the one with highest loss
  - Genetic algorithm(GA)

Evaluation

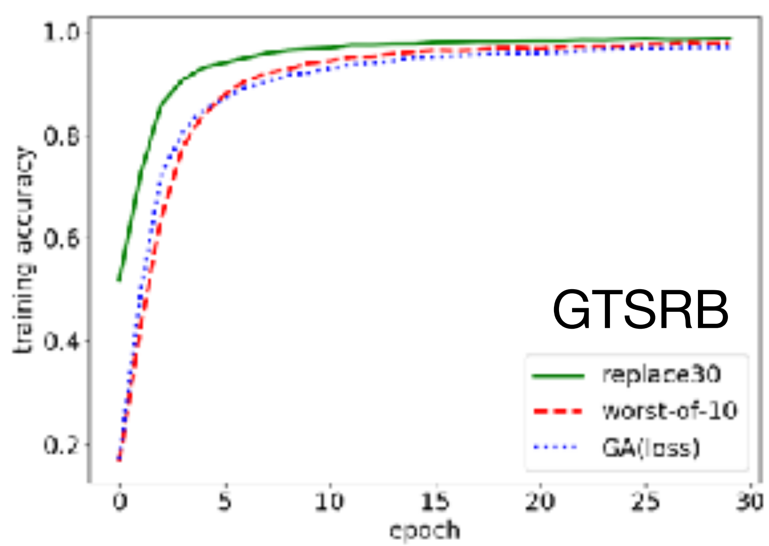
- Attack Space:
    1. - Rotation: rotate image with degree in range [-30, 30]
    2. - Translate: horizontally and vertically shift image at most 10% pixels [-3,3]
    3. - Shear: shear image at most 10% pixels [-0.1, 0.1]
    4. - Zoom: zoom up or zoom down with range [-0.9, 1.1]
    5. - Brightness: change brightness by uniformly adding or subtracting a value for each pixel, the value is in range [-32, 32]
    6. - Contrast: change contrast by scale the RGB value of each pixel with a factor in range [0.8, 1.2]

- Attack Strategy:
    - - Nature: original testing set
    - - Random: perturb original testing set using random perturbation parameters
    - - Grid: perturb using grid parameters, and regard the image is misclassified if one of perturbation is misclassified

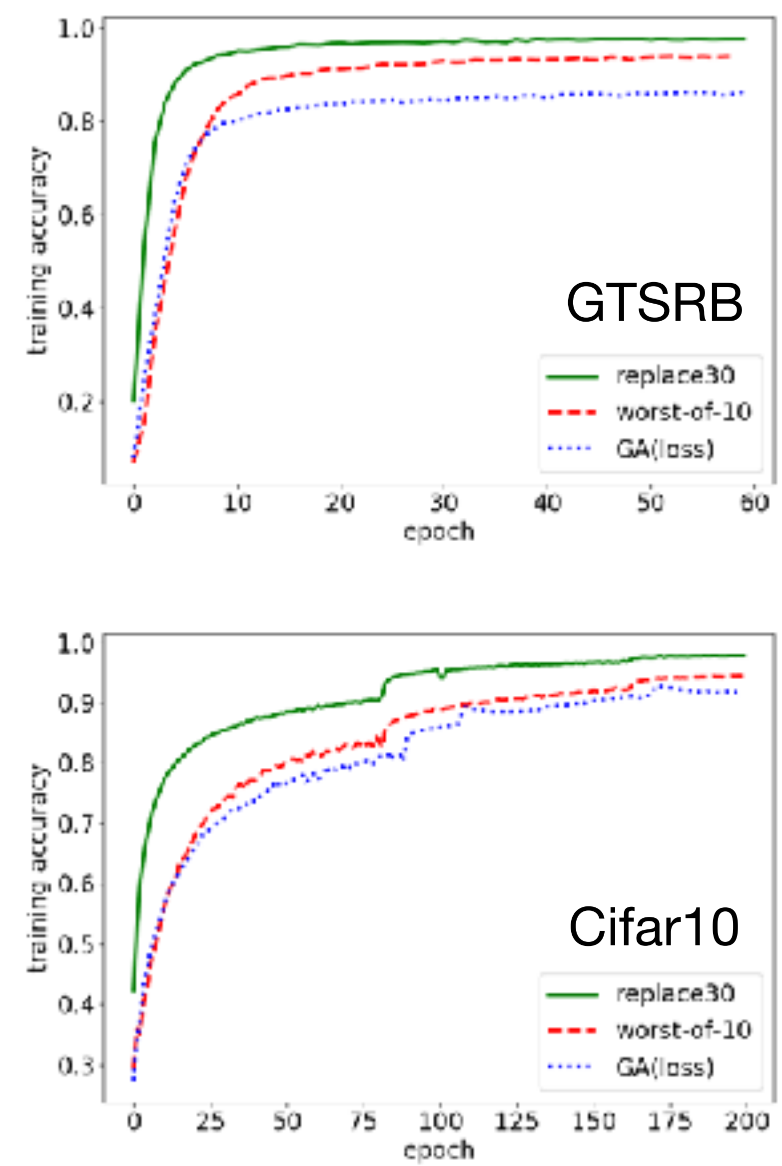
Evaluation

We evaluated the training accuracy based on different augmentation strategies.

- The training accuracy of Genetic algorithm is lower than both replace30 and worst-of-10
- Genetic algorithm is able to find more misclassified perturbations in each step
- GA-based approach is more effective to solve the inner minimization of Saddle point problem



Training accuracy with 3 transformations
(rotate, translate, shear)

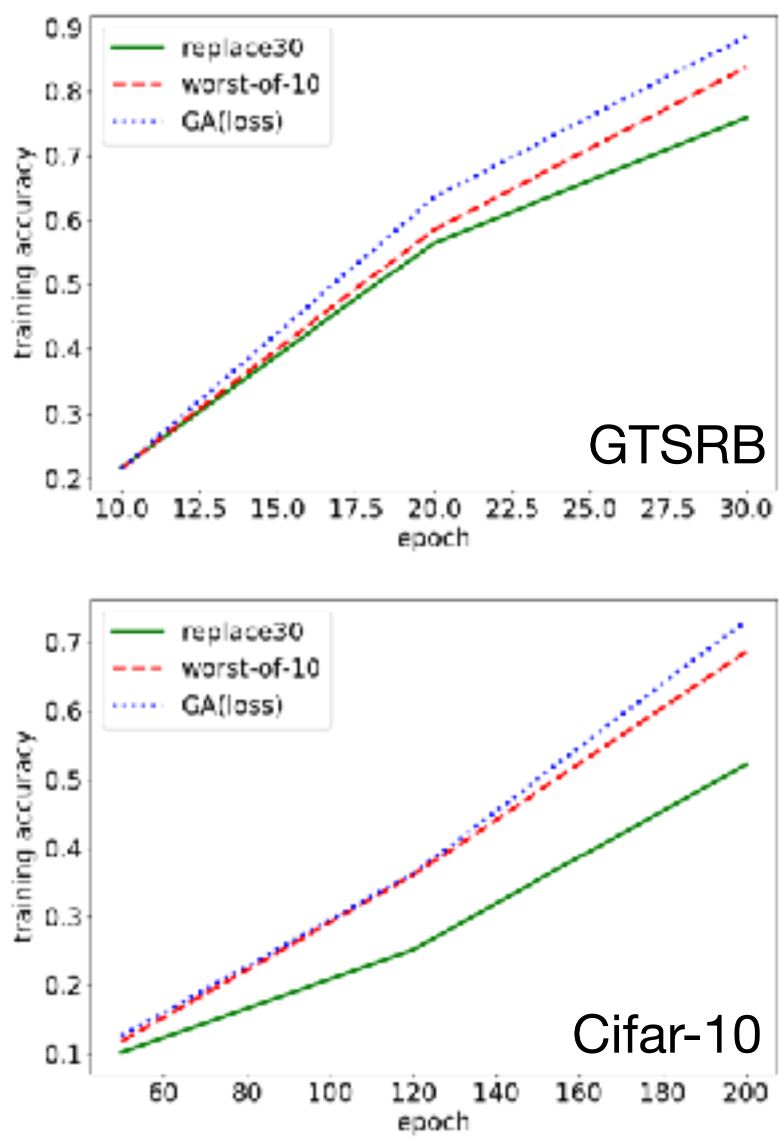


Training accuracy with 6 transformations
(rotate, translate, shear, zoom, brightness, contrast)

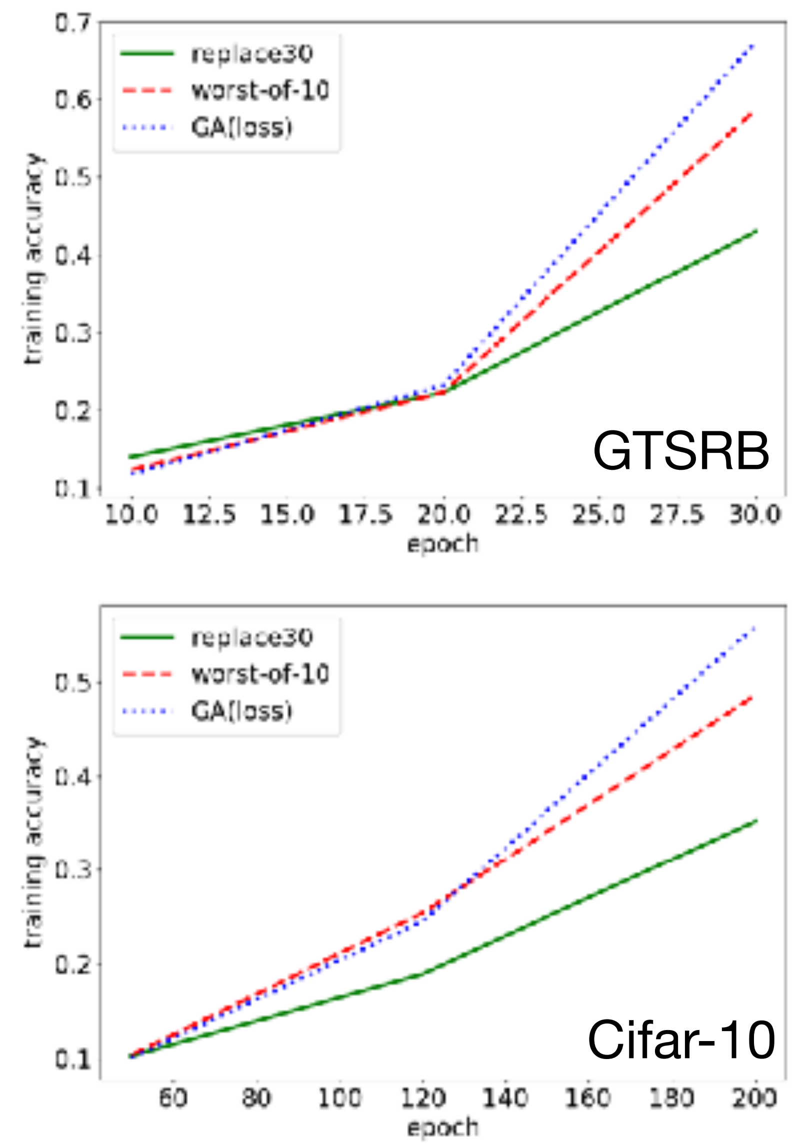
Evaluation

We evaluated the testing accuracy based on grid attack.

- Genetic algorithm is able to generate highest testing accuracy under the grid attack.



Testing accuracy based on grid attack with 3 transformations



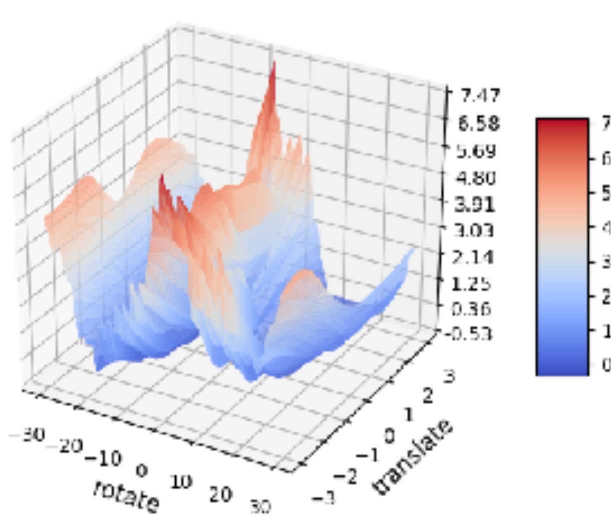
Testing accuracy based on grid attack with 6 transformations

Evaluation - example

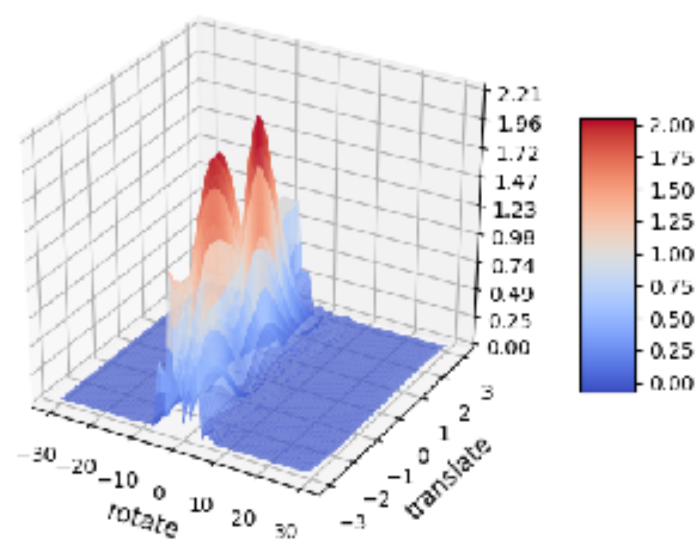
The model trained using GA-based augmentation correctly classifies the following example:

Rotate 1°

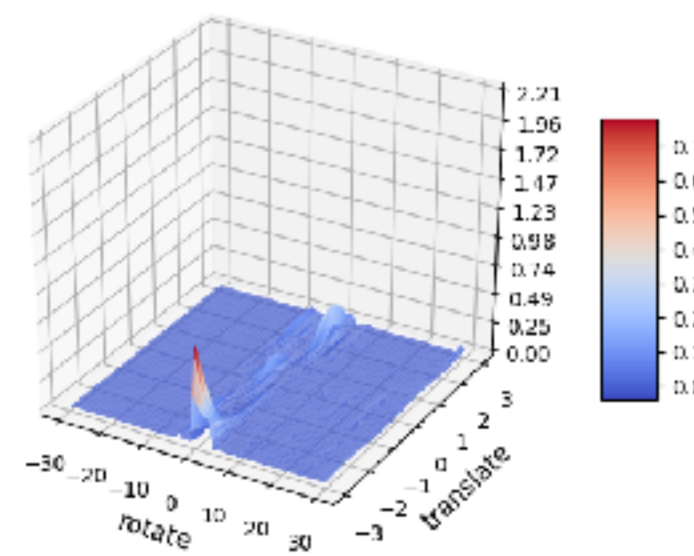
The loss of the perturbed images is significantly reduced by genetic algorithm.



traditional model



worst-of-10 model



GA model

The loss of perturbed images with rotation (-30, 30) and translate (-3, 3)

Raw experimental results for GTSRB

Test accuracy of GTSRB dataset with 3 transformations(rotation, translate, shear).

Aug. strategy	Natural	Random	Grid
Standard	0.979	0.662	0.007
Replace30	0.983	0.978	0.759
Replace40	0.978	0.977	0.795
Worst-of-10	0.980	0.976	0.837
Worst-of-10(cov)	0.983	0.980	0.845
GA(loss)	0.986	0.983	0.888

Test accuracy of GTSRB dataset with 6 transformations.

Aug. strategy	Natural	Random	Grid
Standard	0.973	0.586	0.067
Replace30	0.979	0.956	0.430
Worst-of-10	0.985	0.961	0.586
GA(loss)	0.988	0.972	0.673

Raw experimental results for GTSRB

Test accuracy of GTSRB dataset based on each transformation (the model is trained based on three transformations).

Aug. strategy	Rotation	Translate	Shear
Standard	0.122	0.404	0.939
Replace30	0.901	0.864	0.947
Replace40	0.909	0.846	0.941
Worst-of-10	0.918	0.906	0.946
GA(loss)	0.925	0.921	0.958

Raw experimental results for GTSRB

Average number of misclassified perturbations (totally 81 perturbations for each image). For the model trained using three transformations.

Model	Standard	Rep.30	Rep.40	Worst-of-10	GA
#misclassfied	47.2	3.1	2.67	2.66	1.87

Average number of misclassified perturbations (totally 2187 perturbations for each image). For the model trained using six transformations.

Model	Standard	Rep.30	Worst-of-10	GA
#misclassfied	789	164	117	98

Raw experimental results for Cifar10

Test accuracy of Cifar10 dataset with 3 transformations(rotation, translate, shear).

Aug. strategy	Natural	Random	Grid
Standard	0.875	0.496	0.013
Replace30	0.897	0.897	0.522
Replace40	0.872	0.891	0.626
Worst-of-10	0.892	0.893	0.686
GA(loss)	0.915	0.913	0.732

Test accuracy of Cifar10 dataset with 6 transformations.

Aug. strategy	Natural	Random	Grid
Standard	0.890	0.435	0.011
Replace30	0.901	0.892	0.352
Worst-of-10	0.892	0.893	0.486
GA(loss)	0.915	0.913	0.560

Raw experimental results for Cifar10

Test accuracy of Cifar10 dataset based on each transformation (the model is trained based on three transformations).

Aug. strategy	Rotation	Translate	Shear
Standard	0.093	0.202	0.494
Replace30	0.697	0.651	0.788
Worst-of-10	0.738	0.701	0.811
GA(loss)	0.786	0.740	0.841

Raw experimental results for GTSRB

Average number of misclassified perturbations (totally 81 perturbations for each image). For the model trained using three transformations.

Model	Standard	Rep.30	Worst-of-10	GA
#misclassfied	45.0	10.4	9.8	

Average number of misclassified perturbations (totally 2187 perturbations for each image). For the model trained using six transformations.

Model	Standard	Rep.30	Worst-of-10	GA
#misclassfied	1316	263		232